

## REMARKS

Claims 1 and 3-46 are pending in this application. Claims 1, 3, 4, 8, 14, 44 and 45 were amended herein. Claim 2 was cancelled herein without prejudice or disclaimer. New claim 46 is added herein. Support for the amendments to the claims may be found in the claims as originally filed, particularly in claim 2, and in the specification at page 15, line 23 to page 18, line 5, and page 24, line 2 to page 25, line 1. Support for new claim 46 may be found in claim 1. No new matter has been added. Reconsideration is requested based on the foregoing amendment and the following remarks.

### Interview Summary:

The Applicant submits the following summary of the telephone interview that took place September 16, 2005 between the undersigned representative of the Applicant and the Examiner.

#### Telephone Conference:

The Applicant thanks the Examiner for the many courtesies extended to the undersigned representative of the Applicant during the telephone interview that took place September 16, 2005.

Among the issues discussed during that interview were the amendments to the claims reflected in the Listing of the Claims beginning on page 2 of this paper. The Examiner indicated that the amendments might have placed the claims in condition for allowance, subject to further search and consideration.

### Claim Rejections - 35 U.S.C. § 103:

Claims 1 and 3-45 were rejected under 35 U.S.C. § 103 as being unpatentable over Neches, US 4,445,171 (hereinafter "Neches") in view of Raz et al., US 5,860,137 (hereinafter "Raz"). The rejection is traversed, to the extent it might apply to the claims as amended. Reconsideration is earnestly solicited.

Claim 1 recites,

"a switching request signal detecting section for detecting a switching request signal to request switching such plural processor elements one from another."

Neches neither teaches, discloses, nor suggests switching plural processor elements one from another, let alone "a switching request signal detecting section for detecting a

switching request signal to request switching plural processor elements one from another,” as acknowledged graciously in the Office Action at page 2. The Office Action seeks to compensate for this deficiency of Neches by combining Neches with Raz.

Raz, however, neither teaches, discloses, nor suggest switching plural processor elements one from another either, and thus cannot compensate for the deficiencies of Neches with respect to the claimed invention.

As described in Raz, rather, at column 1, lines 55-58,

In general, in one aspect, the invention is a method of controlling distribution of processing in a system that includes a plurality of host data processors connected to a data storage system.

Thus, Raz is about distributing processing between a plurality of host data processors connected to a data storage system, not “switching plural processor elements one from another either,” as recited in claim 1. All of the of host data processors in Raz will be running, Raz seeks only to equalize their ownership of volumes of a data storage system. In particular, as described in Raz, rather, at column 1, lines 58-65,

The data storage system includes a digital memory that is partitioned into a plurality of volumes. The method includes assigning ownership of the volumes to the host processors such that each of the host processors owns a different subset of volumes. The concept of ownership means that a host processor is prohibited from sending I/O requests to any of the volumes which are outside of the subset of volumes which it owns.

Thus, if Raz is switching anything, he’s switching the ownership of the volumes owned by each of the host processors, not “switching plural processor elements one from another” either, as recited in claim 1.

Furthermore, as described in Raz, rather, at column 1, line 65-column 2, line 6,

The method also includes monitoring the I/O requests that are sent to each of the volumes by each of the host processors; from the monitoring information, generating workload statistics indicative of the distribution of workload among the host processors; detecting a workload imbalance; and in response to detecting the workload imbalance, reassigning ownership of the volumes to the host processors so as to change the distribution of workload among the host processors.

Thus, if Raz is switching anything, he’s switching the ownership of the volumes owned by each of the host processors, not “switching plural processor elements one from another” either, as recited in claim 1.

Thus, even if Neches and Raz were combined, as proposed in the Office Action, the claimed invention would not result.

The Office Action seeks to justify the combination of Neches and Raz at page 3 of the Office Action by saying that,

“It would have been obvious to one skilled in the art at the time of the invention to combine the teachings of Neches and Raz because Raz’s method of switching between processors would improve Neches system by allowing more than one processor to work on a task thereby improving the overall system.”

Neches, however, already allows more than one process to work on a task. Neches, in particular, describes in the Abstract,

A multiprocessor system interouples processors with an active logic network having a plurality of priority determining nodes. Messages are applied concurrently to the network in groups from the processors and are sorted, using the data content of the messages to determine priority, to select a single or common priority message which is distributed to all the processors with a predetermined total network delay time.

Since Neches applies messages *concurrently* to a network of intercoupled multiprocessors, Neches *already* allows more than one process to work on a task. Thus, even if Raz did teach switching plural processor elements one from another either, it is submitted that persons of ordinary skill in the art would have not been motivated to combine Neches with Raz, as proposed in the Office Action, since to do so would have added nothing to Neches.

Claim 1 recites further,

“storing handover information relating to the common program which information is to be handed over from said one processor element to said another processor element.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above. Since neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, they cannot show “storing handover information relating to the common program which information is to be handed over from said one processor element to said another processor element,” as recited in claim 1.

Raz, rather, describes at column 13, lines 1-6,

at said data storage system, receiving a remapping instruction from any one of said plurality of external processors; and  
in said data storage system, in response to receiving said remapping instruction remapping said plurality of volumes to said plurality of connections.

Thus, Raz is describing remapping volumes of a data storage system, not switching plural processor elements one from another, as discussed above. Furthermore, as described in Raz at column 13, lines 26-35,

from information obtained by said monitoring step, generating workload statistics indicative of the distribution of workload among said plurality of external processors;  
detecting a workload imbalance in said workload statistics; and  
in response to detecting said workload imbalance, remapping said plurality of volumes to said plurality of connections.

Thus, Raz is describing remapping volumes of a data storage system, not switching plural processor elements one from another, as discussed above.

Claim 1 recites further,

“a stop control section for stopping the performance of said one processor element after said store control section stores said handover information into said storing section.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above. Since neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, they cannot show “a stop control section for stopping the performance of said one processor element after said store control section stores said handover information into said storing section,” as recited in claim 1.

Finally, claim 1 recites,

“a start control section for starting the performance of said another processor element using said handover information stores in said storing section.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above. Since neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, they cannot show “a start control section for starting the performance of said another processor element using said handover information stores in said storing section,” as recited in claim 1. Claim 1 is submitted to be allowable. Withdrawal of the rejection of claim 1 is earnestly solicited.

Claims 3-43 depend from claim 1 and add additional distinguishing elements. Claims 3-43 are thus also submitted to be allowable. Withdrawal of the rejection of claims 3-43 is earnestly solicited.

Claim 44:

Claim 44 recites,

"a switching request signal detecting section for detecting a switching request signal to request switching such plural processor elements one from another."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "detecting a switching request signal to request switching such plural processor elements, one from another."

Furthermore, persons of ordinary skill in the art would have not been motivated to combine Neches with Raz, as proposed in the Office Action, as also discussed above with respect to claim 1.

Claim 44 recites further,

"storing handover information relating to the common program, which information is to be handed over from said one processor element to said another processor element, into a storing section of said multiprocessor system."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "storing handover information relating to the common program, which information is to be handed over from said one processor element to said another processor element, into a storing section of said multiprocessor system."

Claim 44 recites further,

"after said handover information has been stored into the storing section, stopping the performance of said one processor element."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "after said handover information has been stored into the storing section, stopping the performance of said one processor element"

Finally, claim 44 recites,

"starting the performance of said another processor element using said handover information stored in the storing section."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "starting the performance of said another processor element using said handover information stored in the storing section." Claim 44 ought thus to be allowable as well, for at least those reasons discussed above with respect to claim 1. Withdrawal of the rejection of claim 44 is earnestly solicited.

Claim 45:

Claim 45 recites,

"a switching request signal detecting section for detecting a switching request signal to request switching such plural processor elements one from another."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "detecting a switching request signal to request switching such plural processor elements, one from another."

Furthermore, persons of ordinary skill in the art would have not been motivated to combine Neches with Raz, as proposed in the Office Action, as also discussed above with respect to claim 1.

Claim 45 recites further,

"storing handover information relating to the common program, which information is to be handed over from said one processor element to said another processor element, into a storing section of said multiprocessor system."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "storing handover information relating to the common program, which information is to be handed over from said one processor element to said another processor element, into a storing section of said multiprocessor system."

Claim 45 recites further,

"after said handover information has been stored into the storing section, stopping the performance of said one processor element."

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone "after said handover

information has been stored into the storing section, stopping the performance of said one processor element.”

Finally, claim 45 recites,

“stopping the performance of said another processor element using said handover information stored in said storing section.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone “stopping the performance of said another processor element using said handover information stored in said storing section.” Claim 45 ought thus to be allowable as well, for at least those reasons discussed above with respect to claim 1. Withdrawal of the rejection of claim 45 is earnestly solicited.

New claim 46:

New claim 46 recites,

“a control section for switching between a first processor element and a second processor element during execution of a common program.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone “a control section for switching between a first processor element and a second processor element during execution of a common program.”

New claim 46 also recites,

“a storing section, responsive to each switching of said processor elements by said control section, for storing handover information relating to the execution of the common program.”

Neither Neches nor Raz teach, disclose, or suggest switching plural processor elements one from another, as discussed above with respect to claim 1, let alone “a storing section, responsive to each switching of said processor elements by said control section, for storing handover information relating to the execution of the common program.” New claim 46 is thus believed to be allowable as well, for at least those reasons discussed above with respect to claim 1.

**Conclusion:**

Accordingly, in view of the reasons given above, it is submitted that all of claims 1 and 3-46 are allowable over the cited references. There being no further outstanding objections or rejections, it is submitted that the application is in condition for allowance. An early action to that effect is courteously solicited.

Finally, if there are any formal matters remaining after this response, the Examiner is requested to telephone the undersigned to attend to these matters.

If there are any additional fees associated with filing of this Amendment, please charge the same to our Deposit Account No. 19-3935.

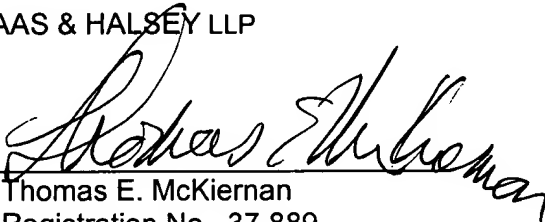
Respectfully submitted,

STAAS & HALSEY LLP

Date:

195605

By:

  
Thomas E. McKiernan  
Registration No. 37,889

1201 New York Avenue, NW, Suite 700  
Washington, D.C. 20005  
Telephone: (202) 434-1500  
Facsimile: (202) 434-1501